Starting with AWX

1- Creating a GitLab SCM Project

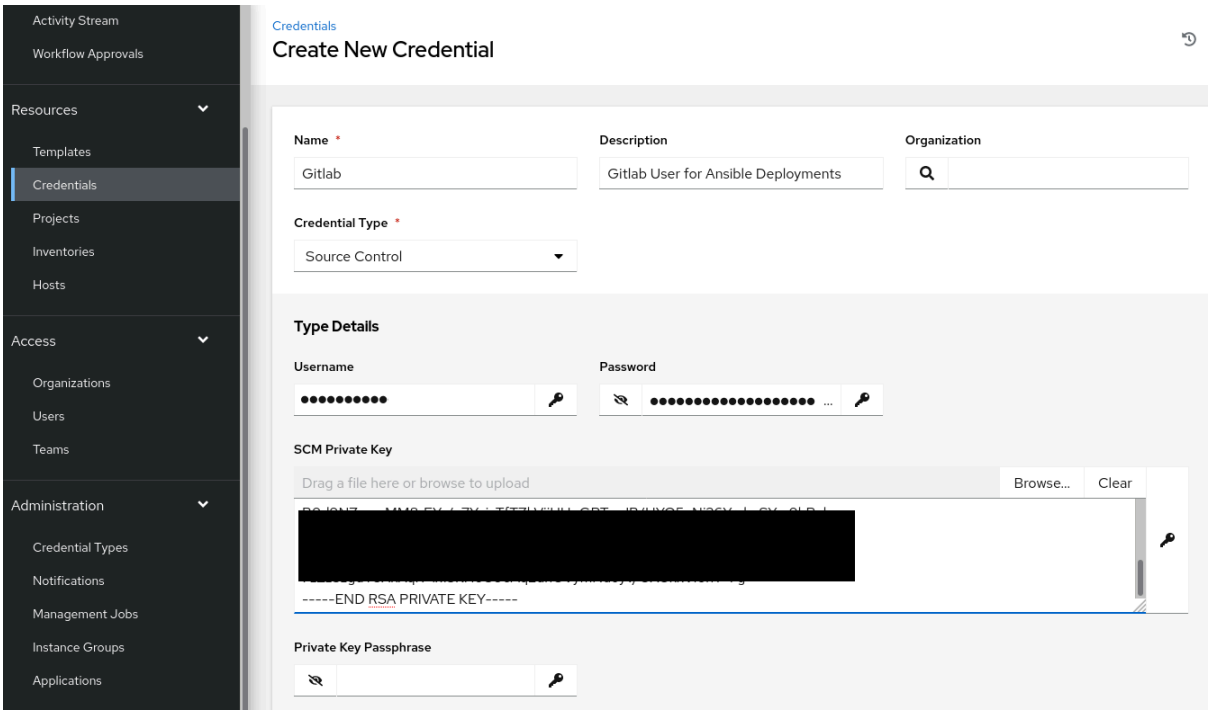
1.1- Gitlab Credentials

To use a Git repository as the source for Ansible Playbooks in Awx, you'll need to add your Gitlab credentials. If you are using a privately hosted Gitlab installation, you will first need to generate an SSH keypair. If you are using the Gitlab hosted service, please refer to the Ansible documentation.

```
cd ~/.ssh
ssh-keygen -N '' -f awx_ssh_key
```

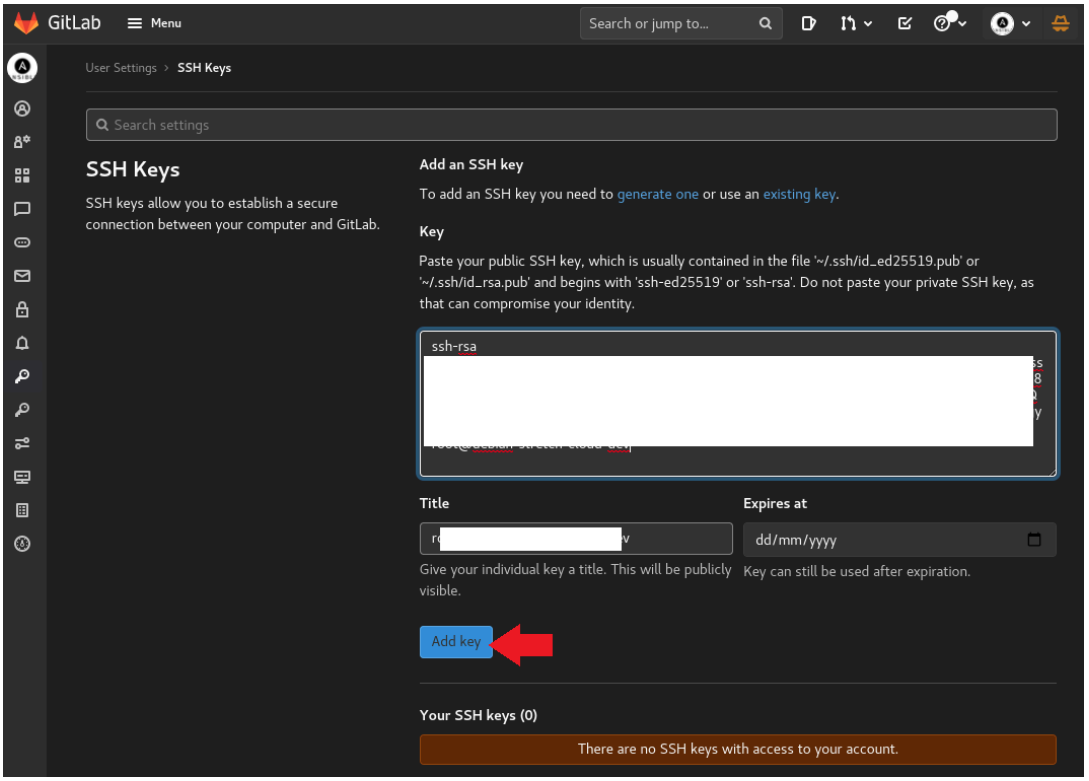
Now add your private key to the AWX Source Control Credentials:

```
cat awx_ssh_key
```



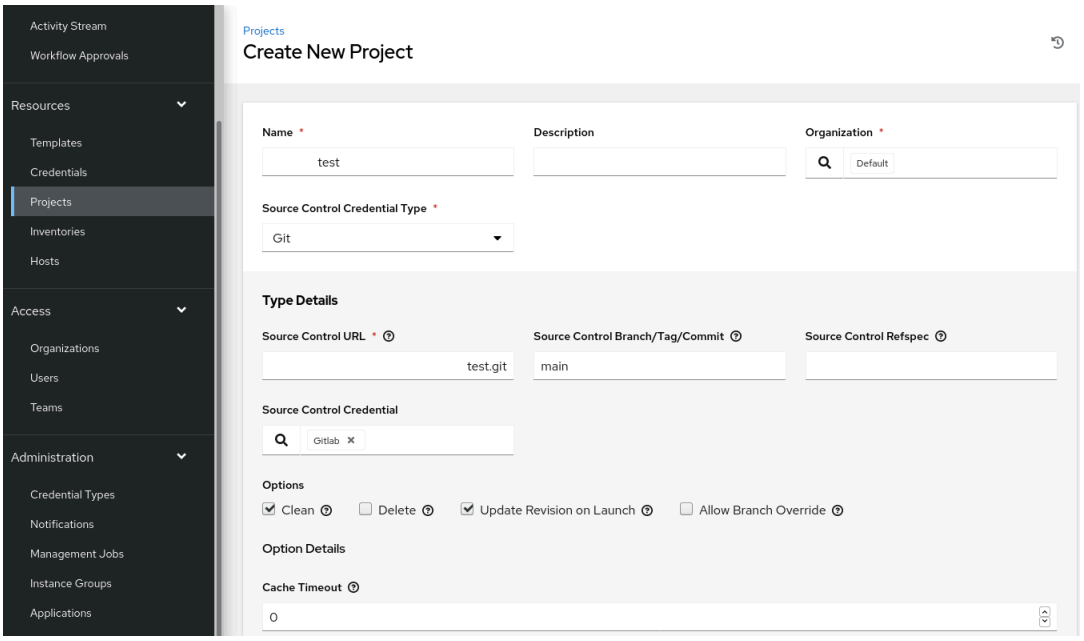
And the public key has to be added to your Gitlab user account:

```
cat awx_ssh_key.pub
```

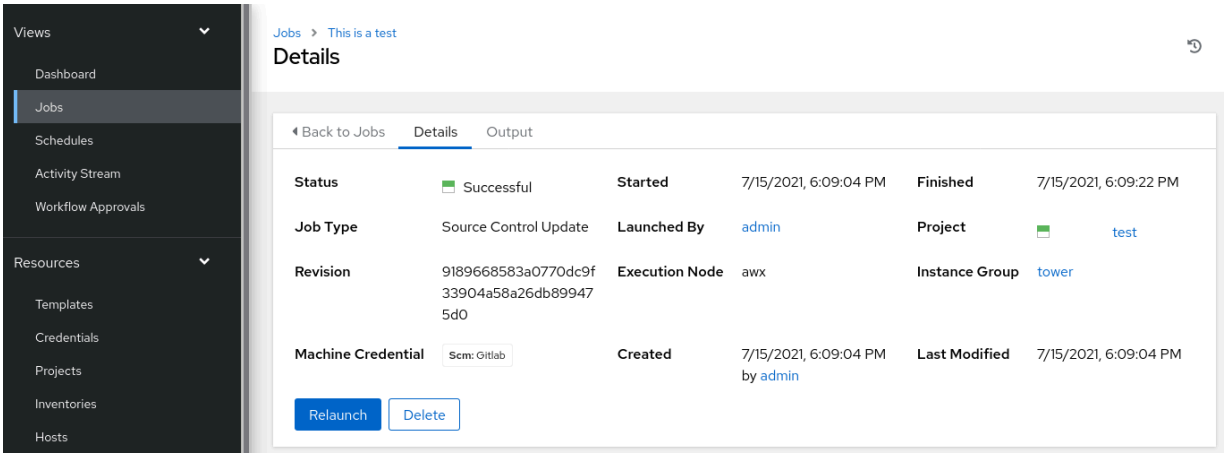


1.2- Creating the Project

The credentials were then used to create a new project. The Source Control URL was set to a newly added test repository on Gitlab.



Save the Project and hit Sync - check under Jobs to see if the run was successful:

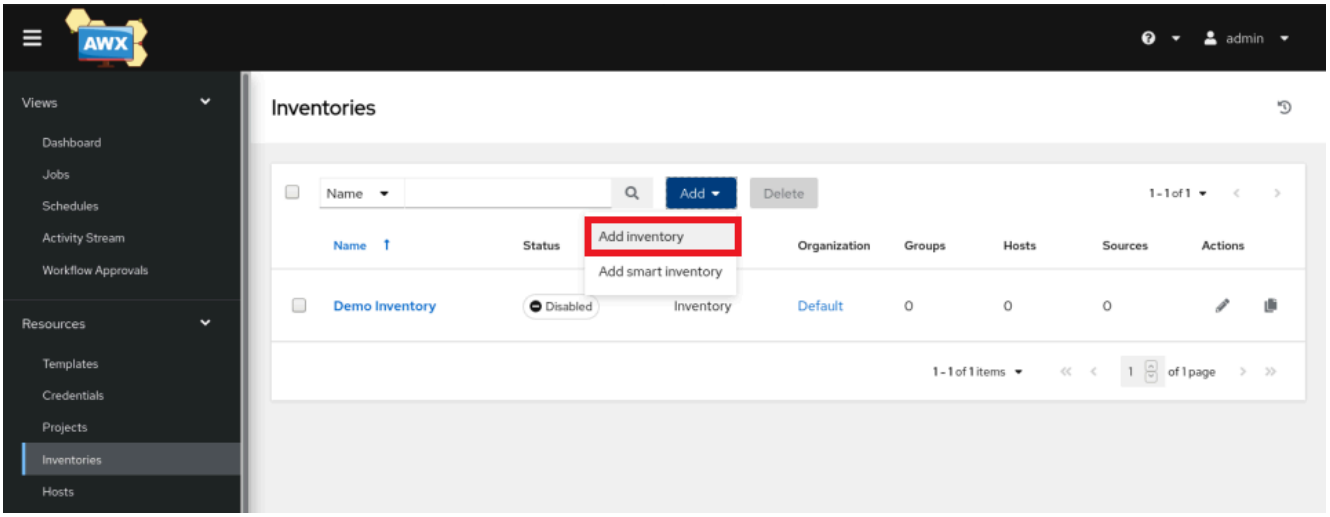


Now that we know that Ansible Tower has access to our Gitlab repo, we can start by building our Ansible Playbook.

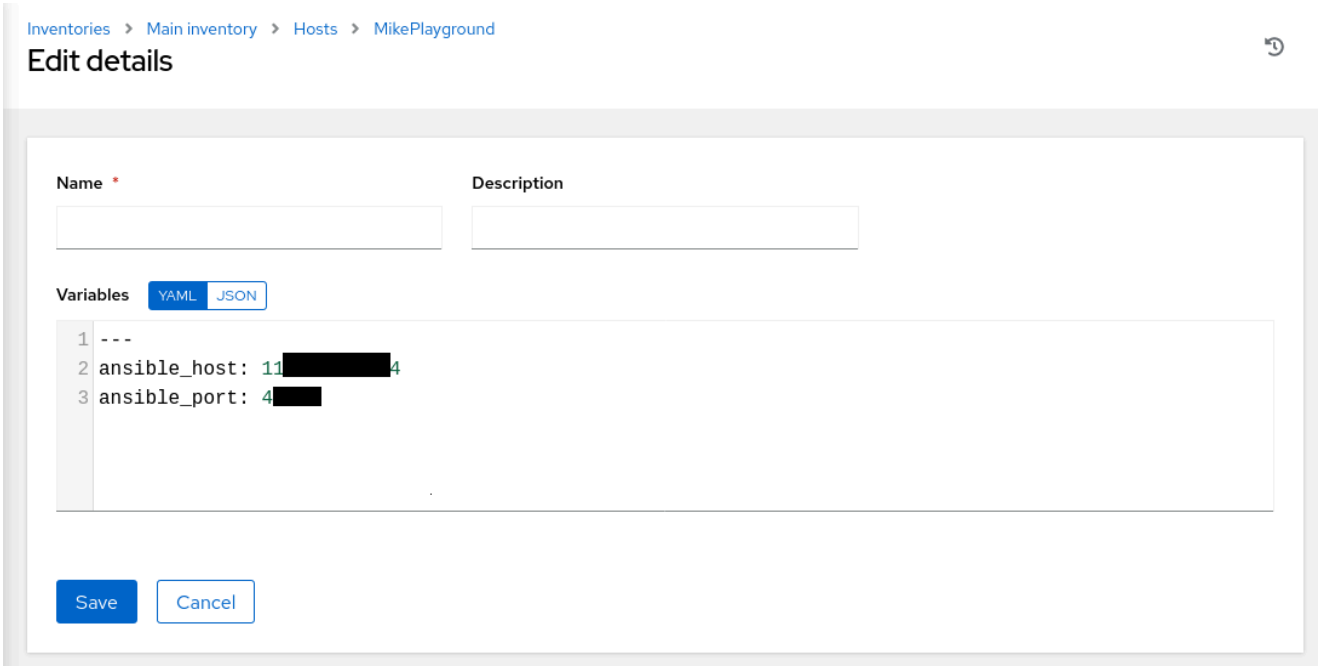
We'll start by a simple debug of a Hello World message for the playbook. This will simply try to connect to our server, write a Hello World string into the terminal and return a successful result if everything worked.

1.3- Adding Hosts

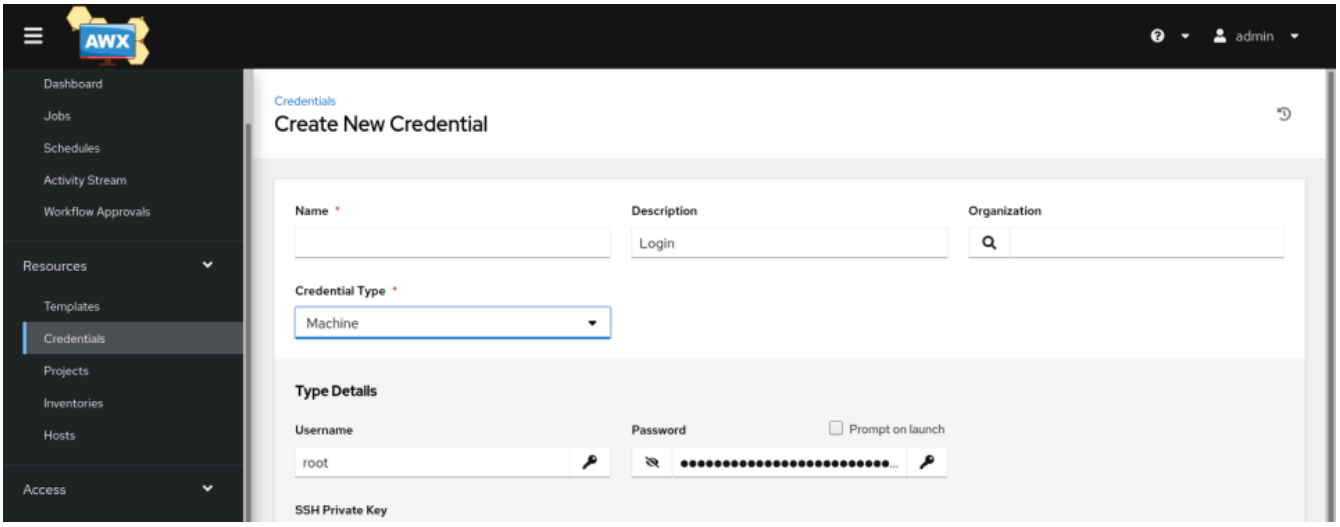
To run our playbook we now have to add a host to Ansible Tower. To add a new host to our AWX frontend we need to create an inventory in the **Resources/Inventories** menu:



After saving the inventory, we can add a host from the Host tab:

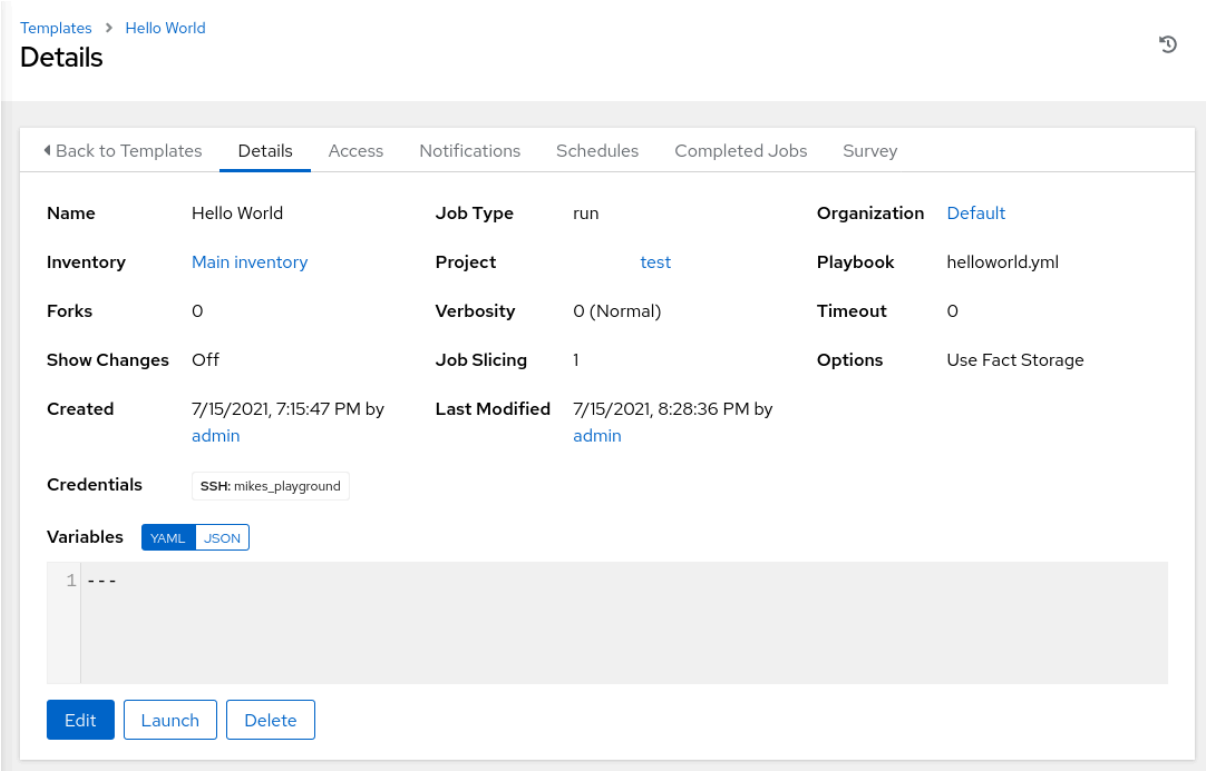


Now we need to switch to the **Resources/Credentials** menu and add our server login:



2- Execute the Playbook

Now that we have all the necessary information for Ansible Tower to work, let's create a Job Template that incorporates everything and test it out.



The job template will be executed once you click "Launch."